Master's Degree Final Project

Master Degree in Industrial Engineering

Control of a high frequency ignition systen based in corona discharge with Arduino

ANNEXS

Author: Albert Baldebey Domènech

Director: Jesús Andrés Álvarez Florez

Date: April, 26th 2017

ETSEIB

Escola Tècnica Superior
d'Enginyeria Industrial de Barcelona

UPC

ANNEXS SUMMARY

A. ARDUINO CODE

This section contains the code that was used in the project and the Arduino library *<PWM.h>* necessary to deliver a high frequency signal.

A.1. Open loop code

This code allows the user to choose the portion of time over 255 bits the signal will be ON with the *dutycycle* variable and the *frequency* of the delivered PWM between 30 Hz 2 MHz. The *pin* 5 of the Arduino board is set to generate it.

```
#include <PWM.h>

//use pin 11 on the Mega instead, otherwise there is a frequency cap at 31 Hz

int pin = 5;              // the pin that the LED is attached to

int dutycycle = 0;        // how bright the LED is

int32_t frequency = 20000; //frequency (in Hz)

void setup()

{  //initialize all timers except for 0, to save time keeping functions

  InitTimersSafe();

  //sets the frequency for the specified pin

  bool success = SetPinFrequency(pin, frequency);

    //if the pin frequency was set successfully, turn pin 13 on

  if(success) {

    pinMode(13, OUTPUT);

    digitalWrite(13, HIGH);

  }

}

void loop()

{
pwmWrite(pin, dutycycle);

dutycycle = 220;

  delay(30);    }
```

A.2. Library

This Arduino library was found in the Internet and it was designed by a user called Sam Knight who made it an open-source code to allow everybody to take advantage of it. It allows the user to deliver, with the Arduino UNO and MEGA boards, PWM signals between 30 Hz and 2MHz.

```
#ifndef PWM_H_

#define PWM_H_

#include "avr/pgmspace.h"

#include "math.h"

#if defined(__AVR_ATmega640__) || defined(__AVR_ATmega1280__) ||
defined(__AVR_ATmega1281__) || defined(__AVR_ATmega2560__) || defined(__AVR_ATmega2561__)

	#include "utility/ATimerDefs.h"

#elif defined(__AVR_ATmega48__) || defined(__AVR_ATmega88__) || defined(__AVR_ATmega88P__)
|| defined(__AVR_ATmega168__) || defined(__AVR_ATmega168P__) || defined(__AVR_ATmega328__)
|| defined(__AVR_ATmega328P__)

	#include "utility/BTimerDefs.h"

#endif



#if defined(__AVR_ATmega640__) || defined(__AVR_ATmega1280__) ||
defined(__AVR_ATmega1281__) || defined(__AVR_ATmega2560__) || defined(__AVR_ATmega2561__)

// 16 bit timers

extern uint32_t GetFrequency_16(const int16_t timerOffset);

extern bool		SetFrequency_16(const int16_t timerOffset, uint32_t f);

extern uint16_t GetPrescaler_16(const int16_t timerOffset);

extern void		SetPrescaler_16(const int16_t timerOffset, prescaler psc);

extern void		SetTop_16(const int16_t timerOffset, uint16_t top);

extern uint16_t GetTop_16(const int16_t timerOffset);

extern void		Initialize_16(const int16_t timerOffset);

extern float	GetResolution_16(const int16_t timerOffset);

// 8 bit timers

extern uint32_t GetFrequency_8(const int16_t timerOffset);
```

```
extern bool        SetFrequency_8(const int16_t timerOffset, uint32_t f);
extern uint16_t GetPrescaler_8(const int16_t timerOffset);
extern void        SetPrescaler_8(const int16_t timerOffset, prescaler psc);
extern void        SetPrescalerAlt_8(const int16_t timerOffset, prescaler_alt psc);
extern void        SetTop_8(const int16_t timerOffset, uint8_t top);
extern uint8_t   GetTop_8(const int16_t timerOffset);
extern void        Initialize_8(const int16_t timerOffset);
extern float      GetResolution_8(const int16_t timerOffset);
#endif
#if defined(__AVR_ATmega48__) || defined(__AVR_ATmega88__) || defined(__AVR_ATmega88P__) || defined(__AVR_ATmega168__) || defined(__AVR_ATmega168P__) || defined(__AVR_ATmega328__) || defined(__AVR_ATmega328P__)
// 16 bit timers
extern uint32_t GetFrequency_16();
extern bool        SetFrequency_16(uint32_t f);
extern uint16_t GetPrescaler_16();
extern void        SetPrescaler_16(prescaler psc);
extern void        SetTop_16(uint16_t top);
extern uint16_t GetTop_16();
extern void        Initialize_16();
extern float      GetResolution_16();
// 8 bit timers
extern uint32_t GetFrequency_8(const int16_t timerOffset);
extern bool        SetFrequency_8(const int16_t timerOffset, uint32_t f);
extern uint16_t GetPrescaler_8(const int16_t timerOffset);
extern void        SetPrescaler_8(const int16_t timerOffset, prescaler psc);
extern void        SetPrescalerAlt_8(const int16_t timerOffset, prescaler_alt psc);
extern void        SetTop_8(const int16_t timerOffset, uint8_t top);
extern uint8_t   GetTop_8(const int16_t timerOffset);
```

```
extern void        Initialize_8(const int16_t timerOffset);

extern float   GetResolution_8(const int16_t timerOffset);

#endif

//common functions

extern void        InitTimers();

extern void        InitTimersSafe();
       //doesn't init timers responsible for time keeping functions

extern void        pwmWrite(uint8_t pin, uint8_t val);

extern void        pwmWriteHR(uint8_t pin, uint16_t val);
    //accepts a 16 bit value and maps it down to the timer for maximum resolution

extern bool        SetPinFrequency(int8_t pin, uint32_t frequency);

extern bool        SetPinFrequencySafe(int8_t pin, uint32_t frequency);   //does not set timers
responsible for time keeping functions

extern float   GetPinResolution(uint8_t pin);
//gets the PWM resolution of a pin in base 2, 0 is returned if the pin is not connected to a timer


#endif /* PWM_H_ */
```

B. DATASHEETS

This section includes the pages of the datasheets of the electronic components that were used during the project: The MOSFET and the toroidal current sensor

B.2. MOSFET RFG50N06

FAIRCHILD SEMICONDUCTOR®

RFG50N06, RFP50N06, RF1S50N06SM

Data Sheet **January 2002**

50A, 60V, 0.022 Ohm, N-Channel Power MOSFETs

These N-Channel power MOSFETs are manufactured using the MegaFET process. This process, which uses feature sizes approaching those of LSI integrated circuits gives optimum utilization of silicon, resulting in outstanding performance. They were designed for use in applications such as switching regulators, switching converters, motor drivers, and relay drivers. These transistors can be operated directly from integrated circuits.

Formerly developmental type TA49018.

Ordering Information

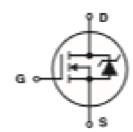
PART NUMBER	PACKAGE	BRAND
RFG50N06	TO-247	RFG50N06
RFP50N06	TO-220AB	RFP50N06
RF1S50N06SM	TO-263AB	F1S50N06

NOTE: When ordering, use the entire part number. Add the suffix, 9A, to obtain the TO-263AB variant in tape and reel, i.e. RF1S50N06SM9A.

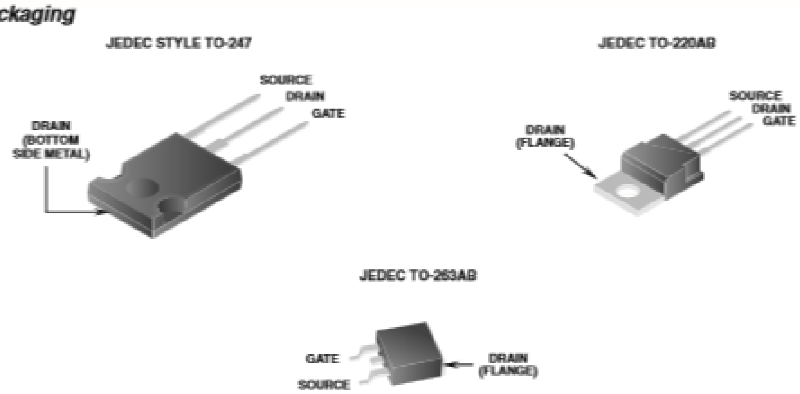
Features

- 50A, 60V
- $r_{DS(ON)} = 0.022\Omega$
- Temperature Compensating PSPICE® Model
- Peak Current vs Pulse Width Curve
- UIS Rating Curve
- 175°C Operating Temperature

Symbol



Packaging



©2002 Fairchild Semiconductor Corporation RFG50N06, RFP50N06, RF1S50N06SM Rev. B

RFG50N06, RFP50N06, RF1S50N06SM

Absolute Maximum Ratings $T_C = 25^oC$, Unless Otherwise Specified

		RFG50N06, RFP50N06 RF1S50N06SM	UNITS
Drain to Source Voltage (Note 1)	V_{DSS}	60	V
Drain to Gate Voltage (R_{GS} = 20kΩ) (Note 1)	V_{DGR}	60	V
Gate to Source Voltage	V_{GS}	±20	V
Continuous Drain Current (Figure 2)	I_D	50	A
Pulsed Drain Current	I_{DM}	(Figure 5)	
Pulsed Avalanche Rating	E_{AS}	(Figure 6)	
Power Dissipation	P_D	131	W
Linear Derating Factor		0.877	W/^{o}C
Operating and Storage Temperature	T_J, T_{STG}	-55 to 175	^{o}C
Maximum Temperature for Soldering			
Leads at 0.063in (1.6mm) from Case for 10s	T_L	300	^{o}C
Package Body for 10s, see Techbrief 334	T_{pkg}	260	^{o}C

CAUTION: Stresses above those listed in "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress only rating and operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied.

NOTE:

1. $T_J = 25^oC$ to 150^oC.

Electrical Specifications $T_C = 25^oC$, Unless Otherwise Specified

PARAMETER	SYMBOL	TEST CONDITIONS		MIN	TYP	MAX	UNITS
Drain to Source Breakdown Voltage	BV_{DSS}	I_D = 250μA, V_{GS} = 0V (Figure 11)		60	-	-	V
Gate to Source Threshold Voltage	$V_{GS(TH)}$	$V_{GS} = V_{DS}$, I_D = 250μA (Figure 10)		2	-	4	V
Zero Gate Voltage Drain Current	I_{DSS}	V_{DS} = 60V, V_{GS} = 0V	$T_C = 25^oC$	-	-	1	μA
			$T_C = 150^oC$	-	-	50	μA
Gate to Source Leakage Current	I_{GSS}	V_{GS} = ±20V		-	-	±100	nA
Drain to Source On Resistance	$r_{DS(ON)}$	I_D = 50A, V_{GS} = 10V (Figures 9)		-	-	0.022	Ω
Turn-On Time	t_{ON}	V_{DD} = 30V, I_D = 50A, R_L = 0.6Ω, V_{GS} = 10V, R_{GS} = 3.6Ω (Figure 13)		-	-	95	ns
Turn-On Delay Time	$t_{d(ON)}$			-	12	-	ns
Rise Time	t_r			-	55	-	ns
Turn-Off Delay Time	$t_{d(OFF)}$			-	37	-	ns
Fall Time	t_f			-	13	-	ns
Turn-Off Time	t_{OFF}			-	-	75	ns
Total Gate Charge	$Q_{g(TOT)}$	V_{GS} = 0 to 20V	V_{DD} = 48V, I_D = 50A, R_L = 0.96Ω, $I_{g(REF)}$ = 1.45mA (Figure 13)	-	125	150	nC
Gate Charge at 10V	$Q_{g(10)}$	V_{GS} = 0 to 10V		-	67	80	nC
Threshold Gate Charge	$Q_{g(TH)}$	V_{GS} = 0 to 2V		-	3.7	4.5	nC
Input Capacitance	C_{ISS}	V_{DS} = 25V, V_{GS} = 0V f = 1MHz (Figure 12)		-	2020	-	pF
Output Capacitance	C_{OSS}			-	600	-	pF
Reverse Transfer Capacitance	C_{RSS}			-	200	-	pF
Thermal Resistance Junction to Case	$R_{\theta JC}$	(Figure 3)		-	-	1.14	^{o}C/W
Thermal Resistance Junction to Ambient	$R_{\theta JA}$	TO-247		-	-	30	^{o}C/W
		TO-220, TO-263		-	-	62	^{o}C/W

Source to Drain Diode Specifications

PARAMETER	SYMBOL	TEST CONDITIONS	MIN	TYP	MAX	UNITS
Source to Drain Diode Voltage	V_{SD}	I_{SD} = 50A	-	-	1.5	V
Reverse Recovery Time	t_{rr}	I_{SD} = 50A, dI_{SD}/dt = 100A/μs	-	-	125	ns

 RFG50N06, RFP50N06, RF1S50N06SM Rev. B

ETSEIB

RFG50N06, RFP50N06, RF1S50N06SM

Typical Performance Curves Unless Otherwise Specified (Continued)

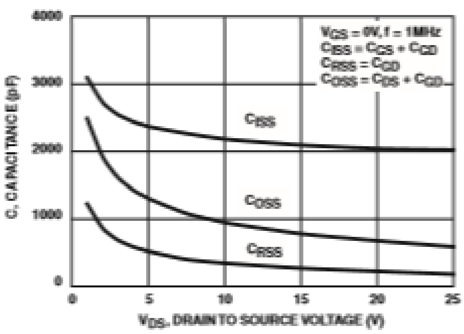
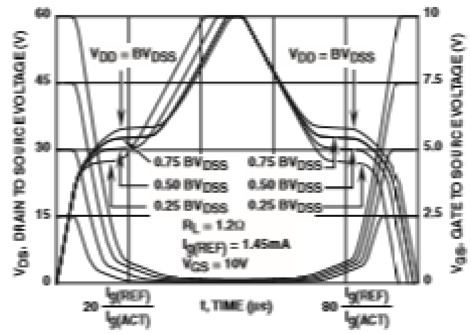


FIGURE 12. CAPACITANCE vs DRAIN TO SOURCE VOLTAGE



NOTE: Refer to Fairchild Application Notes AN7254 and AN7260.

FIGURE 13. NORMALIZED SWITCHING WAVEFORMS FOR CONSTANT GATE CURRENT

Test Circuits and Waveforms

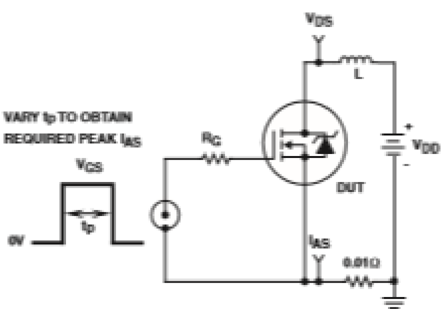


FIGURE 14. UNCLAMPED ENERGY TEST CIRCUIT

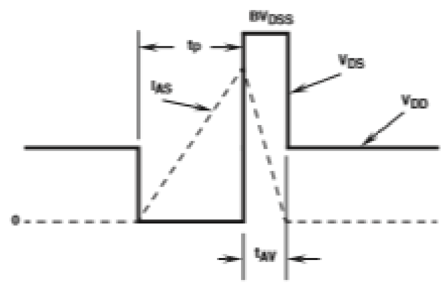


FIGURE 15. UNCLAMPED ENERGY WAVEFORMS

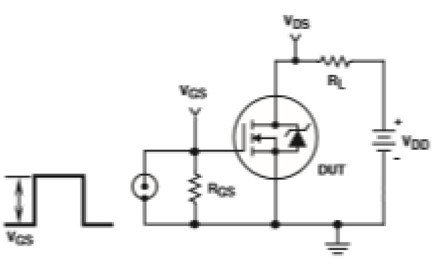


FIGURE 16. SWITCHING TIME TEST CIRCUIT

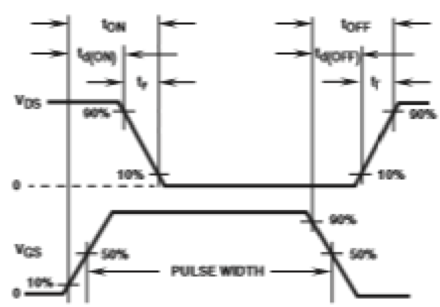
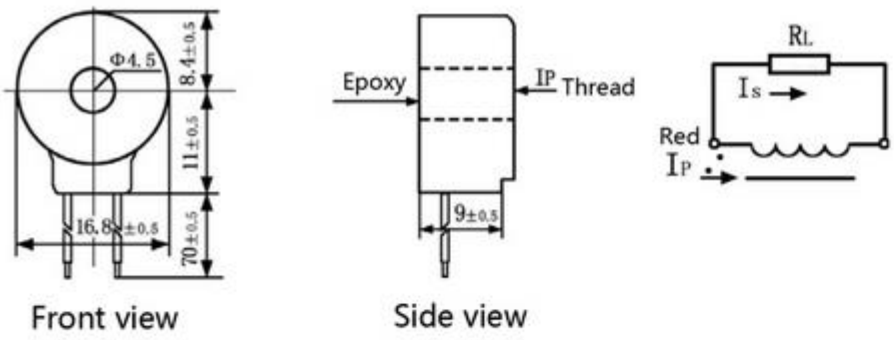


FIGURE 17. SWITCHING WAVEFORMS

B.2. Toroidal current sensor TA12L-100



Technical Parameter											
Model	Ratio	Input Currenton	Output Current	Sample Resistance	Sampling Voltage	Nonlinear Scale	Phase Drift	Working Frequency	Working Temperature	Anti-electricity Level
TA12-100	1000:1	0~5A	0~5mA	200Ω	1V	≤0.3%	≤10′	20Hz~20KHz	-55℃~+85℃	6KV AC/1min
TA12-200	2000:1	0~5A	0~2.5mA	800Ω	2V	≤0.2%	≤5′	20Hz~20KHz	-55℃~+85℃	6KV AC/1min
TA12L-100	1000:1	0~5A	0~5mA	200Ω	1V	≤0.3%	≤10′	20Hz~20KHz	-55℃~+85℃	6KV AC/1min
TA12L-200	2000:1	0~5A	0~2.5mA	800Ω	2V	≤0.2%	≤5′	20Hz~20KHz	-55℃~+85℃	6KV AC/1min